**Questions**

1. What are the important features of JDK 8?
2. What is JVM and is it platform independent?
3. What is the difference between JDK, JVM and JRE?
4. Which class is the super class of all the classes?
5. Why Java does not support multiple inheritance?
6. Why Java is not pure object-oriented language?
7. Difference between overloading and overriding? Can we overload main method?
8. What are different access modifiers and explain with examples?
9. What is the difference between access specifier and access modifier?
10. What is the difference between final and static keyword?
11. Can we declare a class as static?
12. What is the difference between abstract classes and interfaces?
13. What is wrapper classes?
14. What is anonymous inner class?
15. What is ternary operator and explain with example?
16. Explain super, this, garbage collection, pass by value, pass by reference?
17. Write a program to print duplicate elements of an array?
18. Write a program to check whether string is palindrome or not?
19. Write a program to print triangle?
20. Write a program to print series of prime number till 100?

**Answers**

Write a program to print series of prime number till 100

**package** Test;

**class** Prime100

{

**int** i,j;

**public** **void** acc() {

System.***out***.print("prime numbers are ");

**for**(i=3;i<=100;i++)

{

**for**(j=2;j<=i;j++)

{

**if**(i%j==0)

**break**;

}

**if**(i==j)

System.***out***.print(i+", ");

}

}

**public** **static** **void** main(String ar[])

{

Prime100 ob=**new** Prime100();

ob.acc();

}

}

Write a program to print triangle

**package** Test;

**public** **class** Triangle {

**int** i, j, k=4;

**public** **void** patt() {

**for**(i=0;i<5;i++) {

**for**(j=0;j<k;j++) {

System.***out***.print(" ");

}

k=k-1;

**for**(j=0;j<=i;j++) {

System.***out***.print("\* ");

}

System.***out***.println(" ");

}

}

**public** **static** **void** main(String[] args) {

Triangle ob=**new** Triangle();

ob.patt();

}

}

Write a program to check whether string is palindrome or not

**package** Test;

**import** java.util.Scanner;

**class** StringPalindrome

{

**int** i, length;

String str, rev="";

Scanner sc = **new** Scanner(System.***in***);

**public** **void** acc() {

System.***out***.println("enter string:");

str=sc.next();

length=str.length();

**for** (i=length-1; i>=0; i--)

rev=rev + str.charAt(i);

**if** (str.equals(rev))

System.***out***.println(str+" is a palindrome");

**else**

System.***out***.println(str+" is not a palindrome");

}

**public** **static** **void** main(String args[])

{

StringPalindrome ob=**new** StringPalindrome();

ob.acc();

}

}

Write a program to print duplicate elements of an array

**package** Test;

**import** java.util.Scanner;

**public** **class** DuplicateArray

{

**int** i, j;

**int** num[]= **new** **int**[6];

Scanner sc=**new** Scanner(System.***in***);

**public** **void** details()

{

System.***out***.println("enter numbers: ");

**for**(i=0;i<num.length;i++) {

num[i]=sc.nextInt();

}

System.***out***.println("entered numbers: ");

**for**(**int** i:num)

{

System.***out***.println(i);

}

System.***out***.println();

**for**(i=0;i<num.length;i++) {

**for**(j=i+1;j<num.length;j++) {

**if**(num[i]==num[j]) {

System.***out***.println(num[j]+" is duplicate");

}

}

}

}

**public** **static** **void** main(String[] args) {

DuplicateArray ob=**new** DuplicateArray();

ob.details();

}

}

**Explain super, this, garbage collection, pass by value, pass by reference**

Super: The super keyword in java is a reference variable which is used to refer immediate parent class object.

This: Keyword THIS is a reference variable in Java that refers to the current object.

Garbage collection: It allows developers to create new objects without worrying explicitly about memory allocation and deallocation, because the garbage collector automatically reclaims memory for reuse.

Pass by value: It allows us to create new objects without worrying explicitly about memory allocation and deallocation, because the garbage collector automatically reclaims memory for reuse.

Pass by reference: An alias or reference to the actual parameter is passed to the method, that’s why it’s called pass by reference.

**What is ternary operator and explain with example**

Java ternary operator is the only conditional operator that takes three operands. The first operand in java ternary operator should be a Boolean or a statement with Boolean result. If the first operand is **true,**then java ternary operator returns second operand else it returns third operand.

Example:

**public** **class** Ternary {

**int** num1, num2, num3, res;

Scanner sc=**new** Scanner(System.***in***);

**public** **void** details() {

System.***out***.println("enter num1: ");

num1=sc.nextInt();

System.***out***.println("enter num2: ");

num2=sc.nextInt();

System.***out***.println("enter num3: ");

num3=sc.nextInt();

**if**(num1%2==0) {

System.***out***.println("num1 is even");

res=(num1<num2)?num3:num1;

System.***out***.println("true result: "+res);

}

**else** {

System.***out***.println("num1 is odd");

res=(num1>num2)?num1:num3;

System.***out***.println("false result: "+res);

}

}

}

**What is anonymous inner class**

It is an inner class without a name and for which only a single object is created.

**What is wrapper classes**

Java provides wrapper classes where we need to use objects instead of primitive data types.

**What is the difference between abstract classes and interfaces**

|  |  |
| --- | --- |
| Abstract classes can have abstract and non-abstract methods | Interface can have only abstract methods, default and static methods |
| Does not support multiple inheritance | Supports multiple inheritance |

**Can we declare a class as static**

We cannot make the top-class static as the compiler does not allow however, we can make the nested class static.

**What is the difference between final and static keyword**

The main difference between a static and final keyword is that **static** is keyword is used to define the class member that can be used independently of any object of that class. **Final** keyword is used to declare, a constant variable, a method which cannot be overridden and a class that cannot be inherited.

**What is the difference between access specifier and access modifier**

The access specifier determines how accessible the field is to code in other classes.

You can optionally declare a field with a modifier keyword: final or volatile and/or static and/or transient.

**Difference between overloading and overriding Can we overload main method**

Overloading occurs when two or more methods in one class have the same method name but different parameters.

Overriding means having two methods with the same method name and parameters.

Yes, we can overload the main method.

**Why Java is not pure object-oriented language**

Java is not a pure object-oriented language because it supports Primitive datatype such as int, byte, long... etc, to be used, which are not objects.

**Why Java does not support multiple inheritance**

Java doesn’t allow multiple inheritance to **avoid the ambiguity** caused by it.

**Which class is the super class of all the classes**

Object class is the super class of all the classes.

**What is the difference between JDK, JVM and JRE**

**JVM (Java Virtual Machine)** is an abstract machine. It is a specification that provides runtime environment in which java bytecode can be executed.

**JRE** is an acronym for Java Runtime Environment. It is used to provide runtime environment.

**JDK** is an acronym for Java Development Kit. It physically exists. It contains JRE + development tools.

**What is JVM and is it platform independent**

**JVM (Java Virtual Machine)** is an abstract machine. It is a specification that provides runtime environment in which java bytecode can be executed.

JVM is not platform independent, that’s why we have different JVM for different operating systems.

**What are the important features of JDK 8**

Default methods